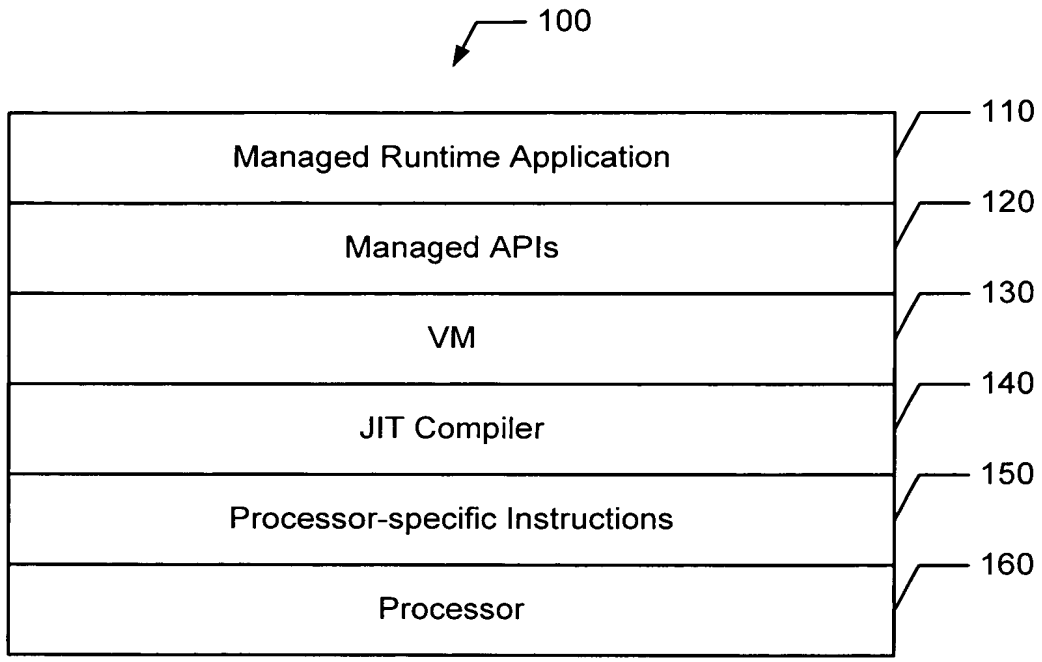

(PRIOR ART)

FIG. 1

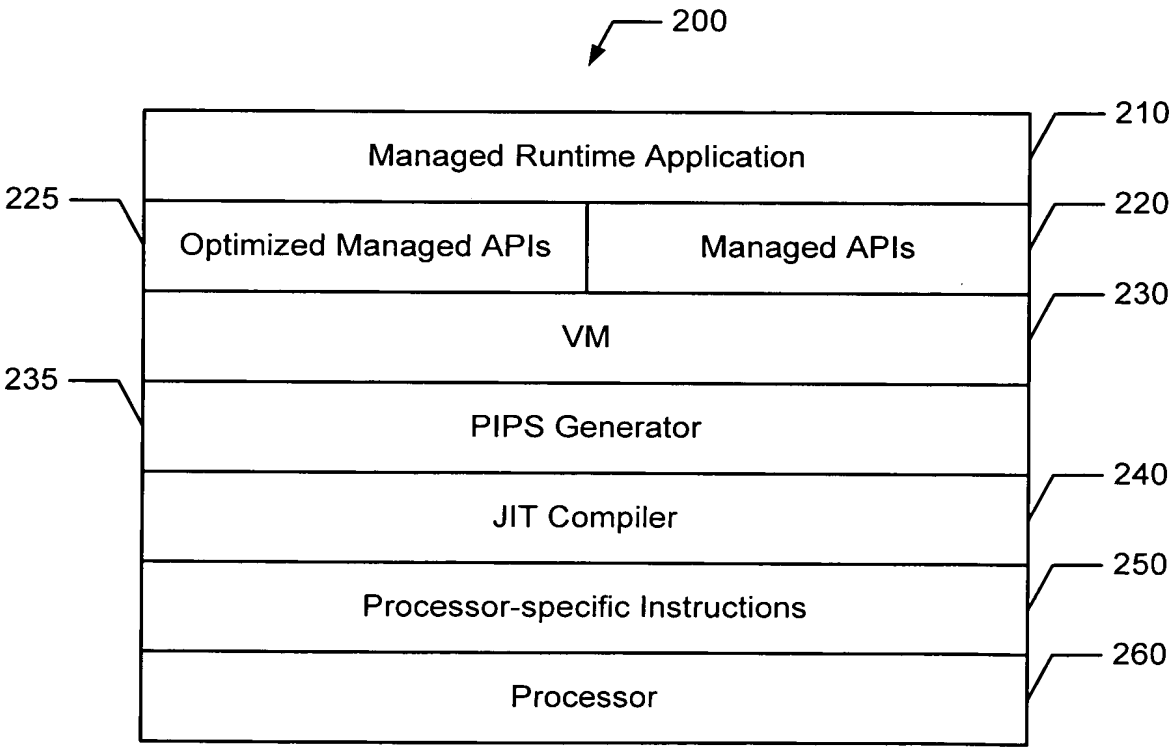

FIG. 2

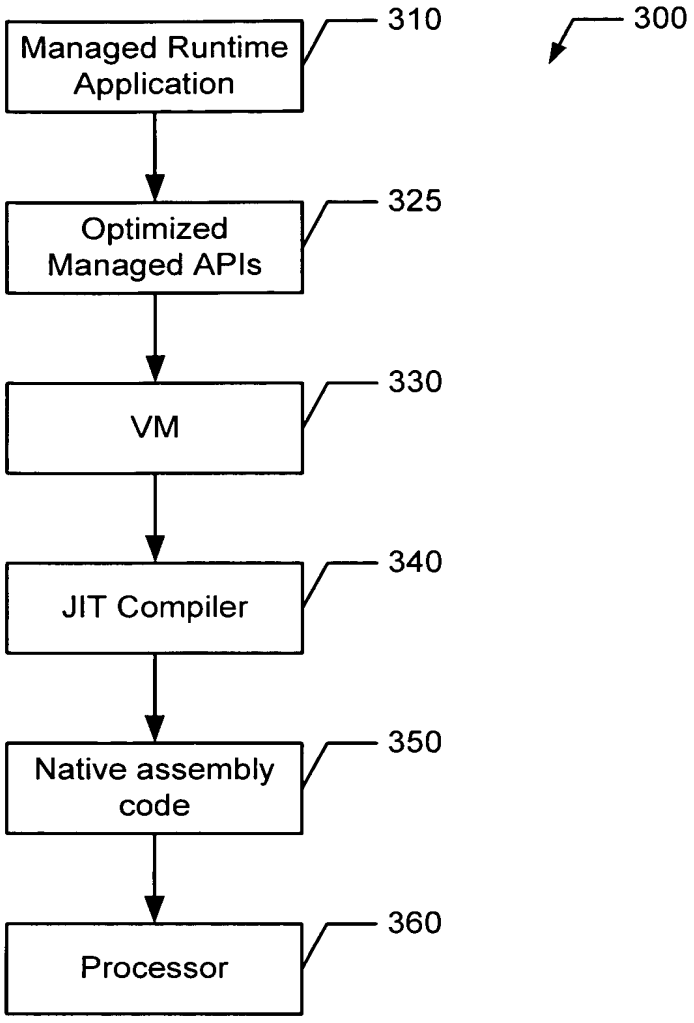

FIG. 3

400

```
LppStatus ownsCompare_16u (const lpp8u* pSrc1, const lpp8u* pSrc2, int len, int *pResult

{
        int i;
        for (i = 0; i < len; i++){
                if (pSrc1[i] != pSrc2[i]) break;
        } // for
        *pResult = (i < len) ? pSrc1[i] - pSrc2[i] : 0;
        return ippStsNoErr;
} // end of function
```

FIG. 4

500

```
Public ownsCompare_16u
_TEXT SEGMENT
pSrc1 EQU 12[esp]
pSrc2 EQU 16[esp]
len EQU 20[esp]
pResult EQU 24[esp]
ALIGN 16
; Lib = W7 (code name for P4 optimization)
; Caller = ippsCompare_16u function
ownsCompare_16u PROC NEAR
        push esi
        push edi
;*******************************************************
;
        mov esi, pSrc1
        mov eax, pSrc2
        mov edi, pSrc2
        mov ecx, len
        test ecx, ecx
        jz ResultCmp16u00
;*******************************************************
;
        xor eax, edi
        and eax, 03h
        jnz ShortLoop4Cmp16u00
        test edi, 01h
        jnz ShortLoop4Cmp16u00
        cmp ecx, 8
        jg Align16Cmp16u00
;*******************************************************
;
SSE2ResultCmp16u00:
        xor eax, 0ffffh
        bsf edx, eax
        lea esi, [esi + edx]
        movzx eax, WORD PTR [esi]
        movzx edx, WORD PTR [esi + edi]
        sub eax, edx
        jmp ResultCmp16u01
;*******************************************************
;
AlignResultCmp16u00:
        sub edi, esi
        jmp SSE2ResultCmp16u00
Result16Cmp16u01:
        add     esi, 16
        jmp     SSE2ResultCmp16u00
Result16Cmp16u02:
        add     esi, 32
        jmp     SSE2ResultCmp16u00
Result16Cmp16u03:
        add     esi, 48
        jmp     SSE2ResultCmp16u00
Result16Cmp16u04:
        add     esi, 64
        jmp     SSE2ResultCmp16u00
Result16Cmp16u05:
        add     esi, 80
        jmp     SSE2ResultCmp16u00
ownsCompare_16u ENDP
_TEXT ENDS
```

510

FIG. 5

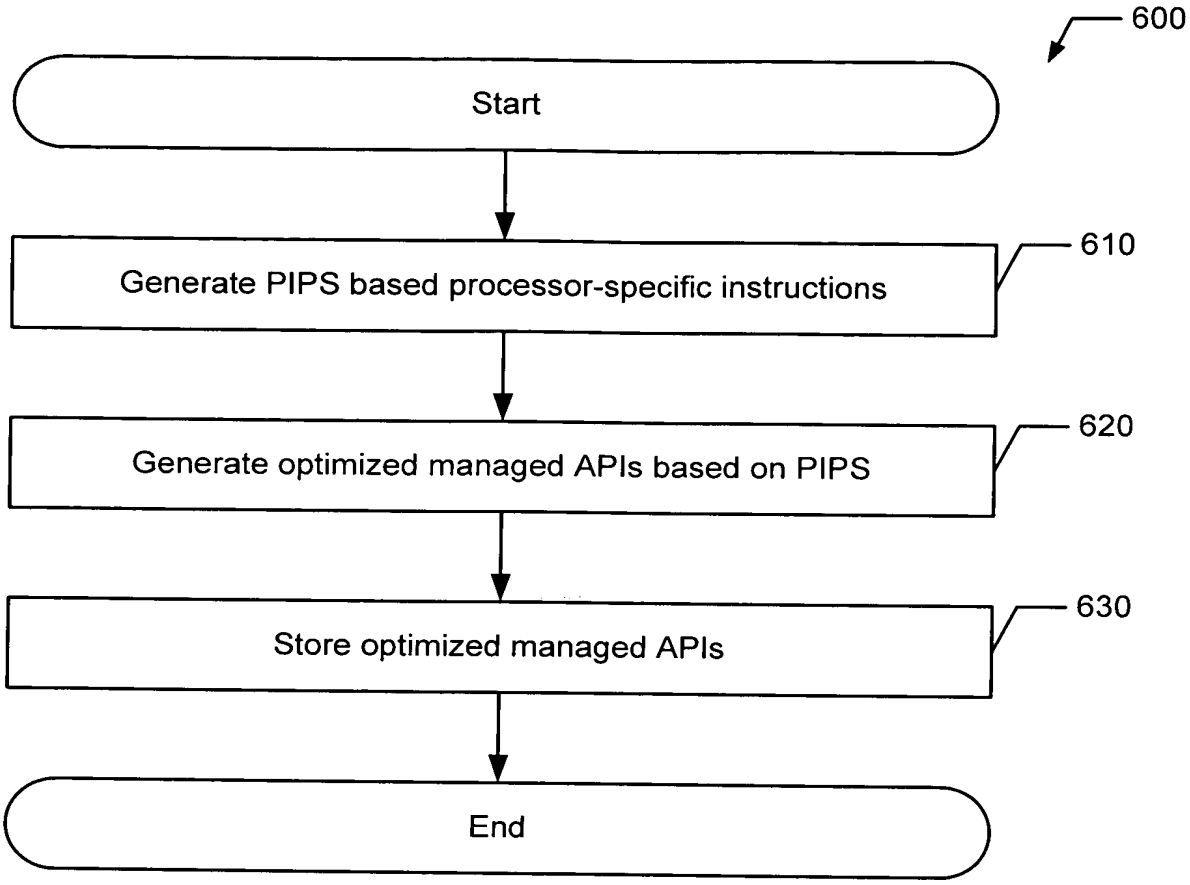
600
Start
Generate PIPS based processor-specific instructions
610
Generate optimized managed APIs based on PIPS
620
Store optimized managed APIs
630
End

FIG. 6

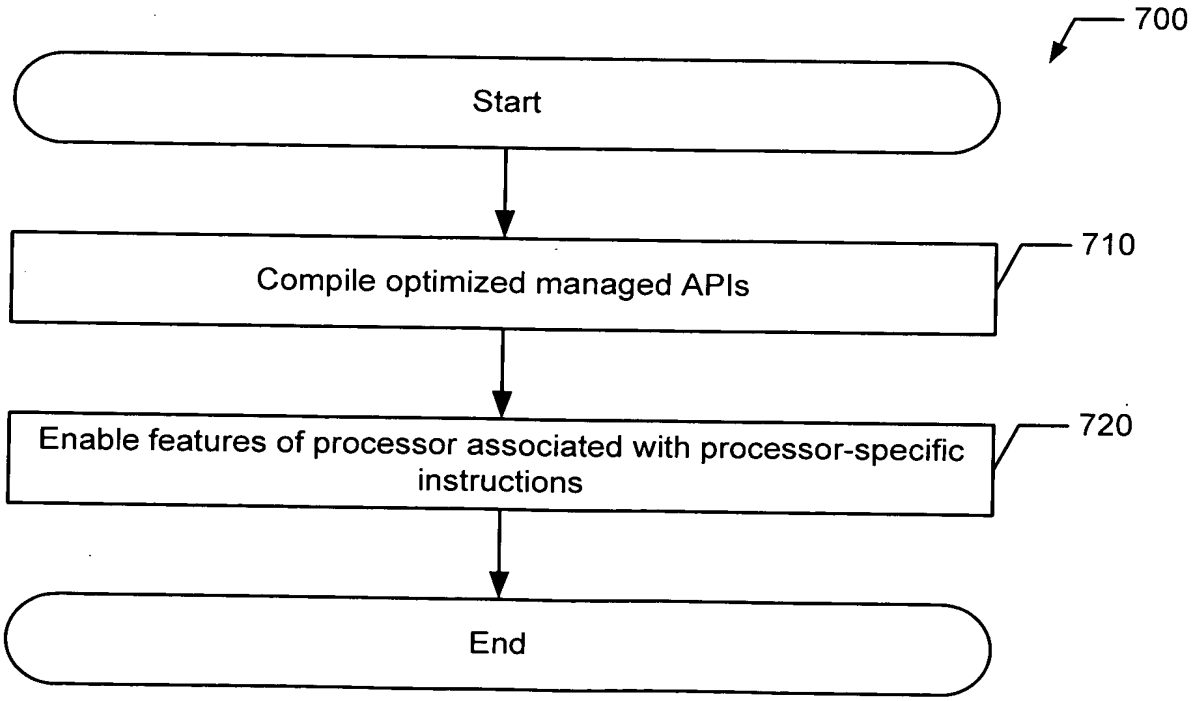
700
Start
Compile optimized managed APIs
710
Enable features of processor associated with processor-specific instructions
720
End

FIG. 7

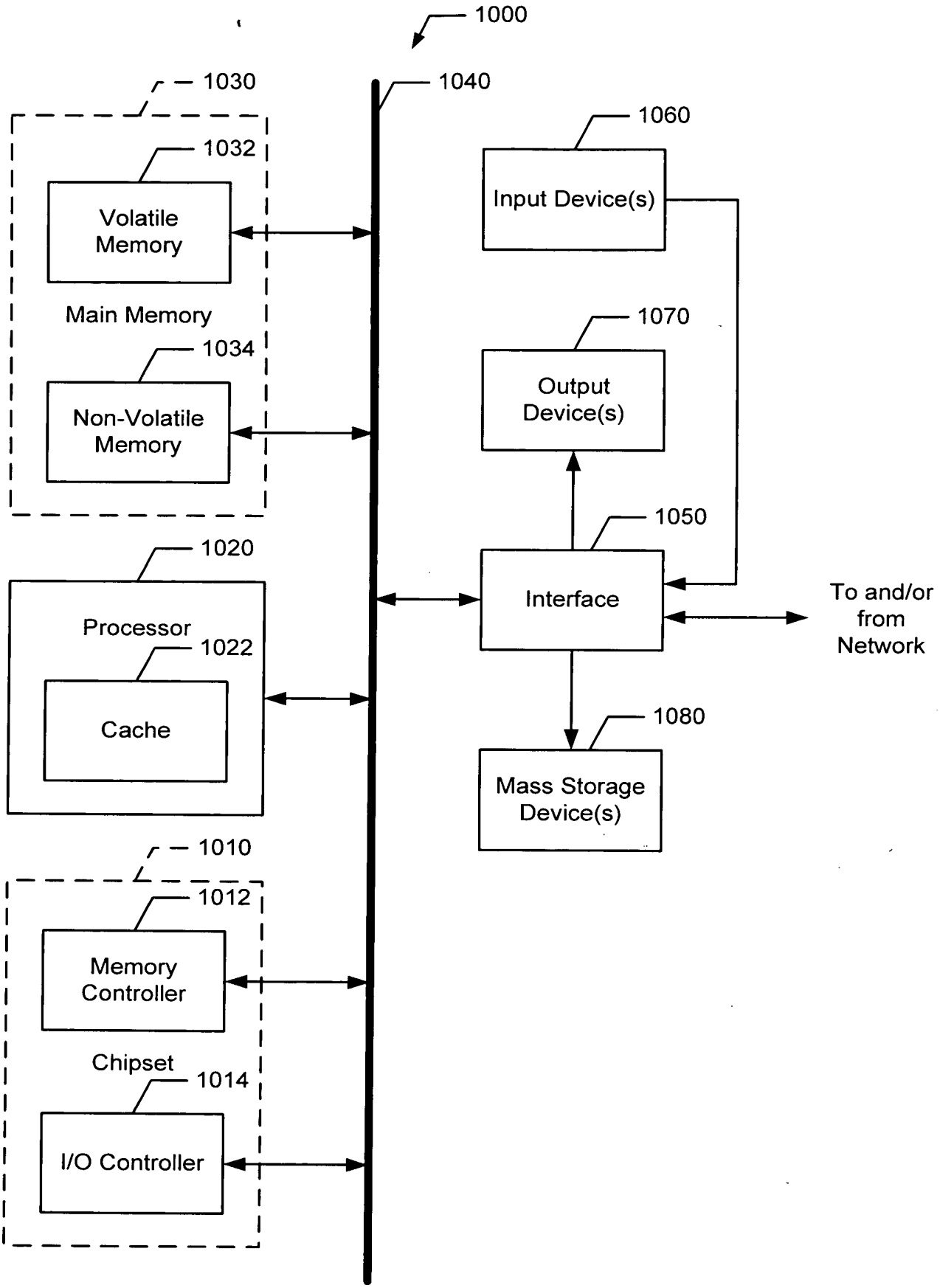
1000
1030
1040
1060
1032
Volatile Memory
Input Device(s)
Main Memory
1070
1034
Output Device(s)
Non-Volatile Memory
1050
1020
Processor
Interface
To and/or from Network
1022
Cache
1080
Mass Storage Device(s)
1010
1012
Memory Controller
Chipset
1014
I/O Controller

FIG. 8